IFW 2661

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
PATENT APPLICATION

Applicant:	Alain Blanc et al.	Art Unit: 2661	
Serial No.:	10/065809	Examiner:	
Filed:	11/21/2002	Atty. Docket:	FR920010071US1

Title:
QUEUE SCHEDULING MECHANISM IN A DATA PACKET TRANSMISSION SYSTEM

Commissioner For Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicants are hereby submitting certified copy of the foreign application, QUEUE SCHEDULING MECHANISM IN A DATA PACKET TRANSMISSION SYSTEM, Patent Application # 01480144.3 filed on 21 DEC 2001, as specified in 35 U.S.C. § 119(b).

Respectfully submitted,

Date: 28 August 2006

By: Anthony J. Canale

Anthony J. Canale, Reg No. 51,526
IP Law Department
IBM Corporation
1000 River Street
Essex Junction, VT 05452
Tel.: 802-769-8782

CERTIFICATE OF MAILING OR FAXING

I, hereby, certify that on the date shown below, this correspondence is being sent by:

MAIL

X deposited with the United States Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner For Patents, P.O. Box 1450, Alexandria VA 22313-1450

FACSIMILE

□ transmitted by facsimile to the Patent and Trademark office

28 August 2006
Date

Arnette Dodge
Name

Arnette Dodge
Signature

Europäisches
Patentamt

European
Patent Office

Office européen
des brevets

Bescheinigung

Certificate

Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. **Patent application No.** **Demande de brevet n°**

01480144.3

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

R C van Dijk

DEN HAAG,DEN
THE HAGUE, 19/03/02
LA HAYE,LE

Blatt 2 der Bescheinigung
Sheet 2 of the certificate
Page 2 de l'attestation

Anmeldung Nr.:
Application no.:
Demande n°:
01480144.3

Anmeldetag:
Date of filing:
Date de dépôt:
21/12/01

Anmelder:
Applicant(s):
Demandeur(s):
INTERNATIONAL BUSINESS MACHINES CORPORATION
Armonk, NY 10504
UNITED STATES OF AMERICA

Bezeichnung der Erfindung:
Title of the invention:
Titre de l'invention:
Queue scheduling mechanism in a packet transmission system

In Anspruch genommene Prioriät(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat:
State:
Pays:

Tag:
Date:
Date:

Aktenzeichen:
File no.
Numéro de dépôt:

Internationale Patentklassifikation:
International Patent classification:
Classification internationale des brevets:
/

Am Anmeldetag benannte Vertragstaaten:
Contracting states designated at date of filing:
Etats contractants désignés lors du depôt:
AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE/TR

Bemerkungen:
Remarks:
Remarques:

QUEUE SCHEDULING MECHANISM IN A PACKET TRANSMISSION SYSTEM

Technical field

The present invention relates to the packet transmission systems wherein the data packets are transmitted from an input device to an output device through a switch engine and relates in particular to a queue scheduling mechanism in such a packet transmission system.

Background

In today's world of telecommunications, characterized by an insatiable demand for bandwidth, there are two particularly very fast growing technology sectors. These are, on one hand, the Internet and, on the second hand, wireless communications. If the first one is primarily concerned with data moving, the second is still mainly dealing with voice, so are the traditional phone carrier service providers. However, all of this is changing very rapidly. Service providers of all types tend to offer more services in an attempt to become, or to stay, profitable. Service offerings range from long distance transport of voice and data over high-speed data backbone to the Internet and data services being offered on wireless pieces of equipment especially wireless phones of second and third generations.

If voice has long been transported under the form of data still, this is on circuit-switched TDM networks which are very different from the Internet packet networks obeying the Internet Protocol (IP). The former is a connection oriented network while the latter is connectionless. Hence, if the first one can offer the carrier-grade type of service required by delay-sensitive applications, such as voice, the second is just of a best effort kind however, well adapted to the transport of data.

All specialized transport network operators want to converge to a same "one-fits-all" type of network i.e. a packet network able to process differently flows of data depending on Quality of Service (QoS) schemes so as flows are indeed processed according to some specific requirements such as delay, jitter, bandwidth, and packet loss.

Switching and routing have been opposed for long in the manner the packets flow through the nodes of the network. The first one tightly associated to connection oriented protocols like ATM requires that a path be established prior to any data movement while routing is essentially the mode of operation of IP, and its hop-by-hop moving of data packets, with a decision to be made at each node. However, the end result is that whichever access protocol is in use, the networks are in actuality becoming switched-packet networks.

When packets arrive in a node, the layer 2 forwarding component of the switching node searches a forwarding table to make a routing decision for each packet. Specifically, the forwarding component examines information contained in the packet's header, searches the forwarding table for a match, and directs the packet from the input interface to the output interface across the switch engine.

A switching node includes generally a plurality of output queues corresponding respectively to the plurality of output adapters and a shared memory for temporarily storing the incoming packets to be switched. The switch architecture is known to potentially provide the best possible performance allowing a full outgoing throughput utilization with no internal blocking and minimum delay.

Every queue is also organized by priority. That is, incoming packet headers, which carry a priority tag, are inspected not only to temporarily store packets in different queues, according to the output ports they are due to leave the switch engine but also are sorted by priority within each queue so that higher priority packets are guaranteed to be admitted first in the shared memory, getting precedence over lower priority traffic. In turn, the switch engine applies the same rule to the admitted packets, always privileging higher priorities. This is achieved by organizing the output queues by priority too. Hence, packet pointers, in each output queues are sorted so that admitted packets of higher priorities exit the switch engine first even though older packets, yet of a lower priority, are still waiting.

Generally, the priorities associated with the data packets are fully preemptive. Thus, if there are 4 priorities from P_0 to P_3, priority P_0 is going to take immediately precedence over any other traffic at priorities P_1-P_3 and so on. This is definitively a feature necessary to be able to handle a mix of voice and real-time traffic along with "pure" data traffic over a single network. This guarantees that data for the former type of applications are handled with no delay so that there is no latency other than the necessary minimum time to traverse the switch engine and, even more importantly, in order that no significant jitter be added to any flow of real-time packets.

However, this is necessarily done at the expense of lower priority traffic which has, in case of congestion, to wait. Even if this is not a problem since the transfer of data files is normally insensitive to delay and jitter, a lower priority e.g. P_3, might be completely starved by higher priorities e.g. P_0-P_2. It is why an improved queue scheduling mechanism disclosed in the European patent application 01480118.7 comprises a credit table providing, at each packet cycle, a value N defining the priority rank to be considered by the queue scheduler whereby a data packet is read by the queue scheduler from the queue device corresponding to the priority N instead of the queue device determined by the normal priority preemption algorithm.

But, in some configurations, it is required by the customer that one or several high priorities may never be preempted by lower priorities despite the drawback mentioned above. It is the case for a communication link transmitting essentially voice or video data.

Summary of the invention

Accordingly, the main object of the intention is to provide a queue scheduling mechanism which avoids a lower priority being starved by higher priorities except one or a few number of higher priorities considered as exhaustive priorities which may never be preempted by lower priorities.

Another object of the invention is to provide a queue scheduling mechanism including both a credit means enabling to give a minimum bandwidth to the lower priority traffic and an exhaustive priority register means registering one or several priorities which may never be preempted.

The invention relates therefore to a queue scheduling mechanism in a packet transmission system comprising at least a transmission device for transmitting data packets, a reception device for receiving the data packets, a set of queue devices respectively associated with a set of priorities each defined by a priority rank for storing each data packet transmitted by the transmission device into the queue device corresponding to its priority rank, a queue scheduler for reading, at each packet cycle, a packet in one of the queue devices, and a credit table providing, at each packet cycle, a value N defining the priority rank to be read by the queue scheduler from the queue device corresponding to the priority N instead of the queue device determined by a normal priority preemption algorithm. The mechanism further comprises an exhaustive priority register means for registering the value of at least one exhaustive priority rank to be read by the queue scheduler from the queue device corresponding to the exhaustive priority rank rather than from the queue device corresponding to the priority N.

Brief description of the drawings

The above and other objects, features and advantages of the invention will be better understood by reading the following more particular description of the invention in conjunction with the accompanying drawings wherein :

- Fig. 1 is a block-diagram representing schematically a switch device wherein a queue scheduling mechanism according to the invention is implemented.
- Fig. 2A and 2B together represent a flow chart showing the steps of the method implemented in the queue scheduling mechanism according to the invention.

Detailed description of the invention

The queue scheduling mechanism according to the invention is, in a preferred embodiment, implemented in a switch engine of a switching node wherein data packets are received from a plurality of input adapters and sent through the switch engine to another plurality of output adapters. However, such a mechanism could be used in any system wherein data packets received from transmitting devices are stored in queues according to several priorities before being read under the control of a queue scheduling mechanism for being sent to receiving devices.

Referring to Fig. 1, a switch engine 10 wherein the invention is implemented, comprises several queue devices 12, 14, 16 and 18 generally organized as FIFOs respectively associated with priority ranks P_0, P_1, P_2 and P_3. This means that data packets having a priority P_0 are stored in queue device 12, data packets of priority P_1 in queue device 14, data packets in priority P_2 in queue device 16, and data packets of priority P_3 in queue device 18.

At each packet cycle, the queue devices 12, 14, 16 and 18 have to be scheduled by a queue scheduler 20 through control lines 21 to allow a data packet to be read and sent to an output adapter 22 wherein the packet is stored in a queue device 24. However, a data packet may be read from a queue device of the switch engine 10 only if a GRANT signal sent on line 26 from the queue device 24 to the queue scheduler 20 is active. The activation of the grant signal for a given priority depends upon an algorithm which is a function of the filling level of queue device 24. Generally, there are several filling thresholds associated respectively with the priority ranks which make the GRANT signal inactive for a priority rank when the threshold associated with this priority rank is reached. Note that a packet of a priority N is read from the corresponding queue device 12, 14, 16 or 18 only if there is at least one packet stored in this queue device. The queue scheduler 20 is aware of this by means of lines 25 from the queue devices.

In order to avoid that a data packet having a low priority stays in the switch engine for a very long time due to highest priority traffic resulting in holding a switch resource preventing highest priority packets to be queued and setting a time out at the end user level followed by a retransmission of the packet which increases the network congestion, the switch engine is also provided with a credit table 28 which enables to guarantee a minimum bandwidth for any priority rank. The credit table 28 which is programmable indicates which priority is allowed to be served unconditionally at each packet cycle, thus overriding the normal preemptive priority mechanism. Such a credit table can be a RAM memory having 256 locations wherein the address to be read is incremented at each packet cycle, the address returning to 0 when it reaches the value 255.

For example, the credit table can be organized according to the hereunder table :

The number of locations containing each value N is defined according to a predetermined percentage of occurrences with respect to the values of the other priority ranks. In the present case, it can be seen that the priority P_3 is registered at addresses 0, 100... that is in 1 location out of 100 locations of the credit table, the priority P_2 is registered at addresses 1, 21, 41, ... that is in 1 location out of 20 locations of the credit table and the priority P_1 is registered at addresses 2, 12, 22 ... that is 1 location out of 10 locations of the credit table, the other address locations being not fulfilled meaning the priority P_0 by default since, in such a case, it is the priority P0 which is served first before the other priorities.

Address	Priority
0	P3
1	P2
2	P1
:	:
12	P1
:	:
21	P2
22	P1
:	:
32	P1
:	:
41	P2
42	P1
:	:
100	P3
:	:

Accordingly, the credit provided to the different priority ranks is the following in percentage :

P0	84%
P1	10%
P2	5%
P3	1%

In order to avoid that a lower priority given by the credit table 28 may preempt the highest priority (P0), or several of the highest priorities, the switch engine is provided with an exhaustive priority register means 30 for registering the exhaustive priorities. At each packet cycle, the exhaustive priority register mans 30 is read by the queue scheduler 20 to determine whether there is a packet having an exhaustive priority which is waiting for being transmitted. It is only when there is no such a packet that a packet of the priority rank pointed in credit table 28 may be transmitted.

The method for implementing the queue scheduling mechanism according to the invention is illustrated by the flow chart of Fig. 2A and 2B together. At each packet cycle, a variable n is set to 0 (step 40). It is then checked whether the GRANT signal is active for priority n (an exhaustive priority) or, in other words, whether there is authorization to send a packet having the priority 0 since n = 0 (step 42). If so, it is determined whether there is a priority 0 packet to be read in the queue corresponding to priority 0 (step 44). If it is the case, a priority 0 packet is read in the corresponding queue and sent to the output device (step 46). Then, the address of the credit table is incremented (step 48) and the process is looped back to step 40.

If the signal GRANT is not active for the priority 0, or if there is no priority 0 packet in the corresponding queue, it is determined whether there are other exhaustive priorities further to the priority 0 such as priority 1, 2 ... (step 50). If so, the variable n is incremented to n+1, i.e. from 0 to 1 in the present example (step 52) and the above processing is repeated in order to send a priority 1 packet. Such a processing is repeated until there is no more exhaustive priority.

Then the credit table is read (step 54) to know the priority rank which is recorded at the address being read at this cycle. It is assumed that the priority rank being recorded is the priority N, N being a number different from 0 as mentioned above or 0 by default. It is then checked whether the GRANT signal is on for this priority, that is whether there is authorization to send a priority N packet (step 56). If so, it is determined whether there is a packet to be read in the queue corresponding to priority N (step 58). If it is the case, a priority N packet is read in the corresponding queue and sent to the output device (step 60). Then, the address of the credit table is incremented (step 48) and the process is looped back to step 40.

If the signal GRANT is not active for the priority N which has been read from the credit table or if there is no priority N packet in the corresponding queue, it is then checked whether there is authorization to send a priority n+1 packet (the GRANT signal is active) for the considered priority (step 62), that is the highest priority after the exhaustive priorities. If so, it is determined whether there is a packet to be read in the queue corresponding to the priority n+1 (step 64). If it is the case, a priority n+1 packet is read from the queue corresponding to this priority and sent to the output device (step 66). Then, the address of the credit table is incremented (step 48) and the process is looped back to step 40.

If the signal GRANT is not active for the priority n+1 or if there is no priority n+1 packet in the corresponding queue, it is checked whether the value of n+1 has reached the value M corresponding to the lowest priority (step 68). If so, the address of the credit table is incremented and the process is looped back to step 40. If it is not the case, variable n is incremented to n+1 (step 70) and the process returns to step 62 of processing the packet of priority n+1, and so on.

It must be noted that, if there are a credit table and an exhaustive priority register in the switch engine as described in reference to Fig. 1 and not in the input adapter and the output adapter, there is a risk that the lower priority packets be not scheduled and stay in the adapter queue as long as there is higher priority traffic. It is therefore necessary that a credit table with the same percentage of the priority ranks (e.g. 1% for P_3, 5% for P_2 and 10% for P_1 as seen above) and an exhaustive priority register recording the same exhaustive priorities exist in the input adapter as well as in the output adapter.

CLAIMS

1. Queue scheduling mechanism in a packet transmission system comprising at least a transmission device for transmitting data packets, a reception device (22) for receiving said data packets, a set of queue devices (12, 14, 16, 18) respectively associated with a set of priorities each defined by a priority rank for storing each data packet transmitted by said transmission device into the queue device corresponding to its priority rank, a queue scheduler (20) for reading, at each packet cycle, a packet in one of said queue devices, and a credit means (28) providing, at each packet cycle, a value N defining the priority rank to be read by said queue scheduler from the queue device corresponding to the priority N instead of the queue device determined by a normal priority preemption algorithm ;

said mechanism being characterized in that it comprises an exhaustive priority register means for registering the value of at least one exhaustive priority rank to be read by said queue scheduler from the queue device corresponding to said exhaustive priority rank rather than from the queue device corresponding to said priority N.

2. Queue scheduling mechanism according to claim 1, wherein said credit means is a credit table (28) storing at each address a value N equal to one of said priority ranks, the address to be read by said queue scheduler (20) for determining said priority N being incremented at each packet cycle.

3. Queue scheduling mechanism according to claim 2, wherein a data packet is read by said queue scheduler (20) from said queue device (12, 14, 16 or 18) corresponding to said exhaustive priority rank only if an active GRANT signal from said reception device (22) is received by said queue scheduler.

4. Queue scheduling mechanism according to claim 3, wherein said GRANT signal depends upon the filling level of a receiving queue device (24) in said reception device (22) into which the data packets read from said queue devices (12, 14, 16 or 18) are stored.

5. Queue scheduling mechanism according to claim 4, wherein a data packet is read from the queue device (12, 14, 16, 18) determined by said priority N when there is no data packet available in the queue device corresponding to said exhaustive priority.

6. Queue scheduling mechanism according to claim 5, wherein a data packet is read from the queue device (12, 14, 16, 18) determined by said normal priority preemption algorithm when there is no data packet available in neither the queue device corresponding to said exhaustive priority rank nor in the queue device corresponding to said priority N.

7. Queue scheduling mechanism according to claim 6, wherein the number of locations of said credit table (28) containing each value N is defined according to a predetermined percentage of occurrences with respect to the values of the other priority ranks.

8. Queue scheduling mechanism according to claim 7, wherein a number of locations in said credit table (28) contain no value meaning that the priority rank to be considered is the highest priority rank.

9. Queue scheduling mechanism according to any one of claims 1 to 8, being used in a switch engine of a switching node within a network wherein said transmission device is an input adapter and said reception device is an output adapter.

QUEUE SCHEDULING MECHANISM IN A PACKET TRANSMISSION SYSTEM

Abstract

Queue scheduling mechanism in a packet transmission system comprising at least a transmission device for transmitting data packets, a reception device (22) for receiving the data packets, a set of queue devices (12, 14, 16, 18) respectively associated with a set of priorities each defined by a priority rank for storing each data packet transmitted by the transmission device into the queue device corresponding to its priority rank and a queue scheduler (20) for reading, at each packet cycle, a packet in one of the queue devices determined by a normal priority preemption algorithm. A credit table (28) provides, at each packet cycle, a value N defining the priority rank to be read by the queue scheduler from the queue device corresponding to the priority N instead of the queue device determined by the normal priority preemption algorithm. The mechanism comprises an exhaustive priority register means (30) for registering the value of at least one exhaustive priority rank to be read by the queue scheduler from the queue device corresponding to the exhaustive priority rank rather than from the queue device corresponding to the priority N.

FIG. 1

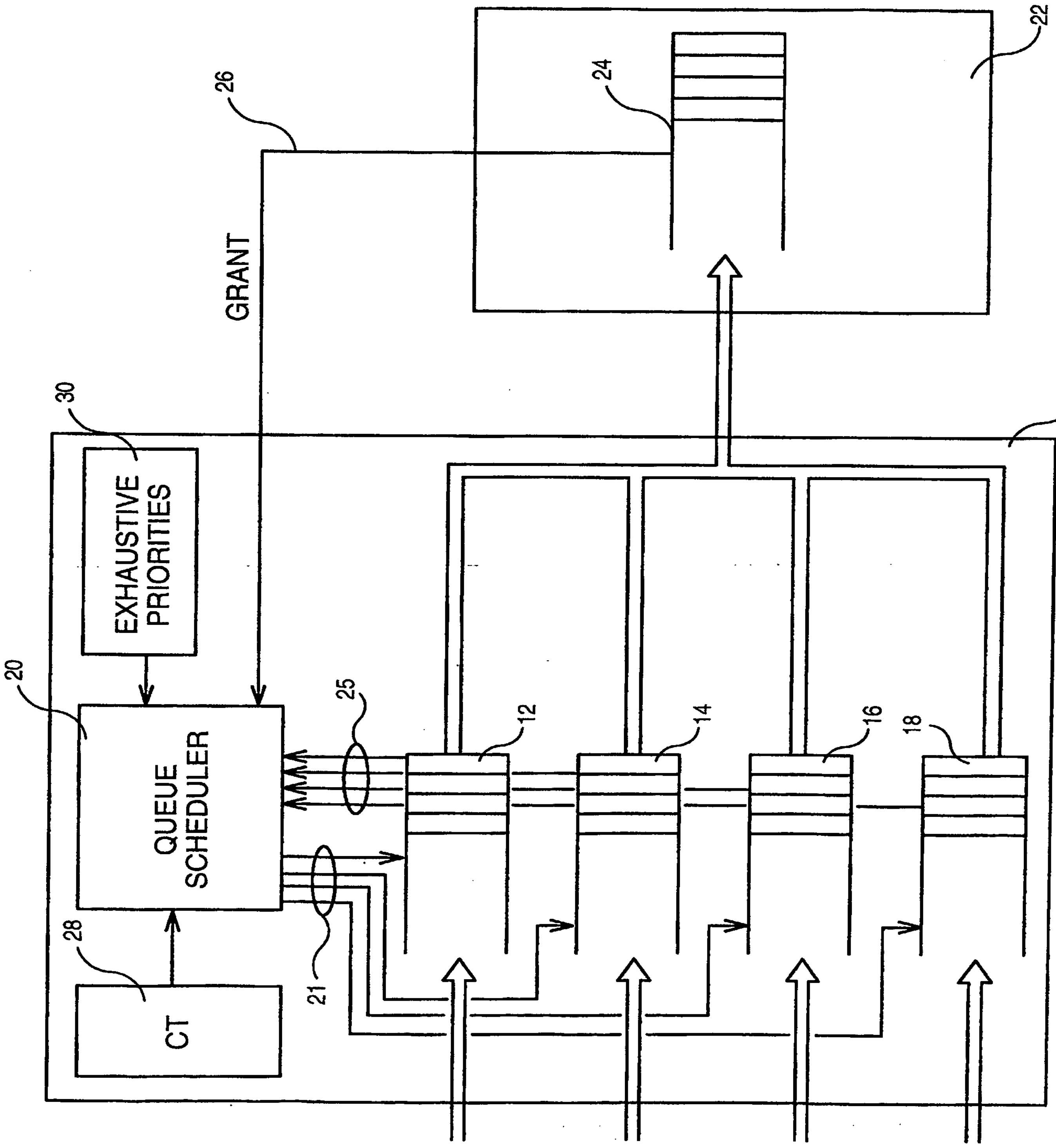
GRANT
26
24
22
30
EXHAUSTIVE PRIORITIES
20
QUEUE SCHEDULER
25
12
14
16
18
10
28
CT
21

FIG. 1

FR 9 2001 0071
Blanc et al
2/3

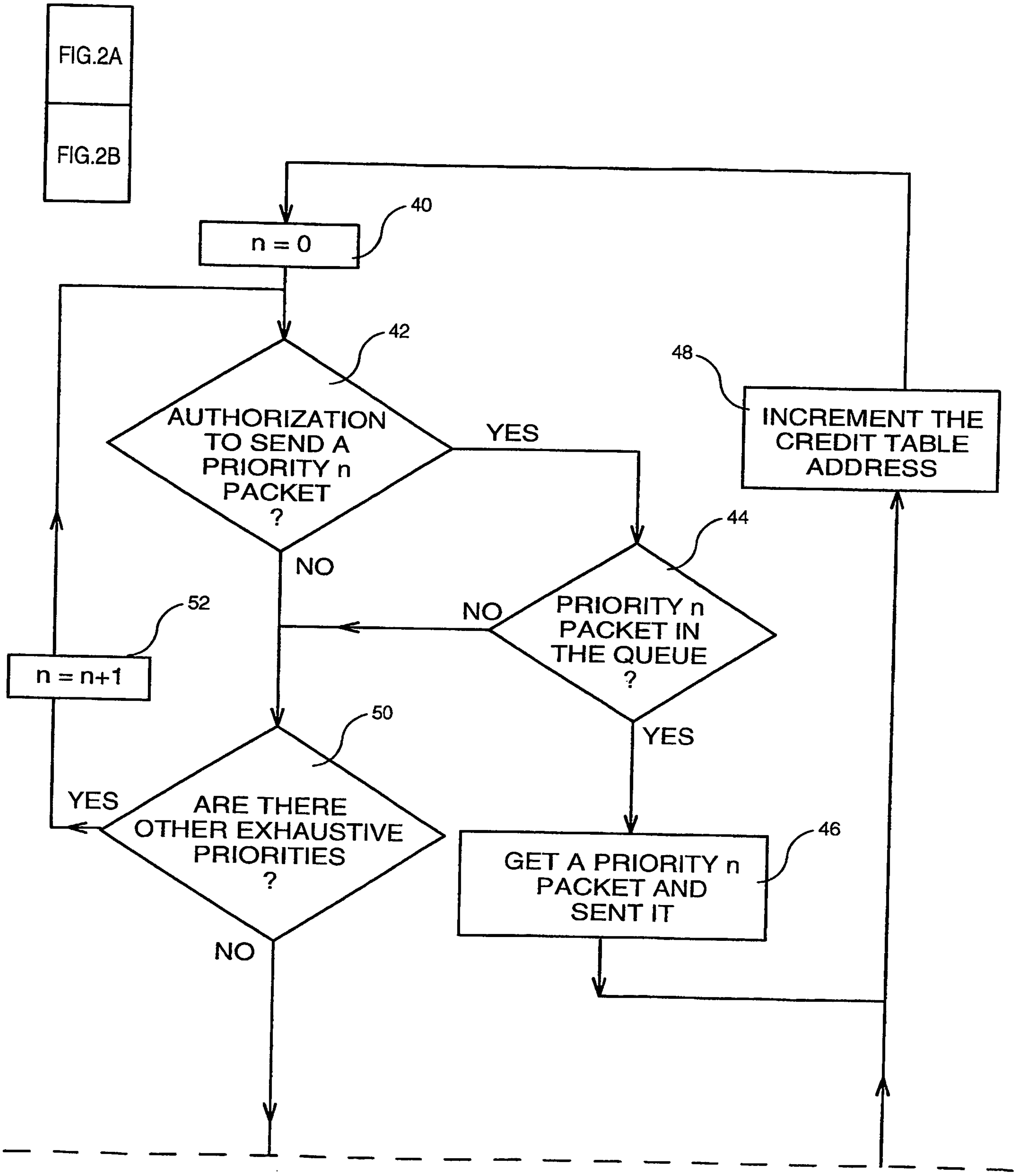
FIG.2A
FIG.2B
40
n = 0
42
AUTHORIZATION TO SEND A PRIORITY n PACKET ?
YES
NO
48
INCREMENT THE CREDIT TABLE ADDRESS
44
PRIORITY n PACKET IN THE QUEUE ?
NO
YES
52
n = n+1
50
ARE THERE OTHER EXHAUSTIVE PRIORITIES ?
YES
NO
46
GET A PRIORITY n PACKET AND SENT IT

FIG. 2A

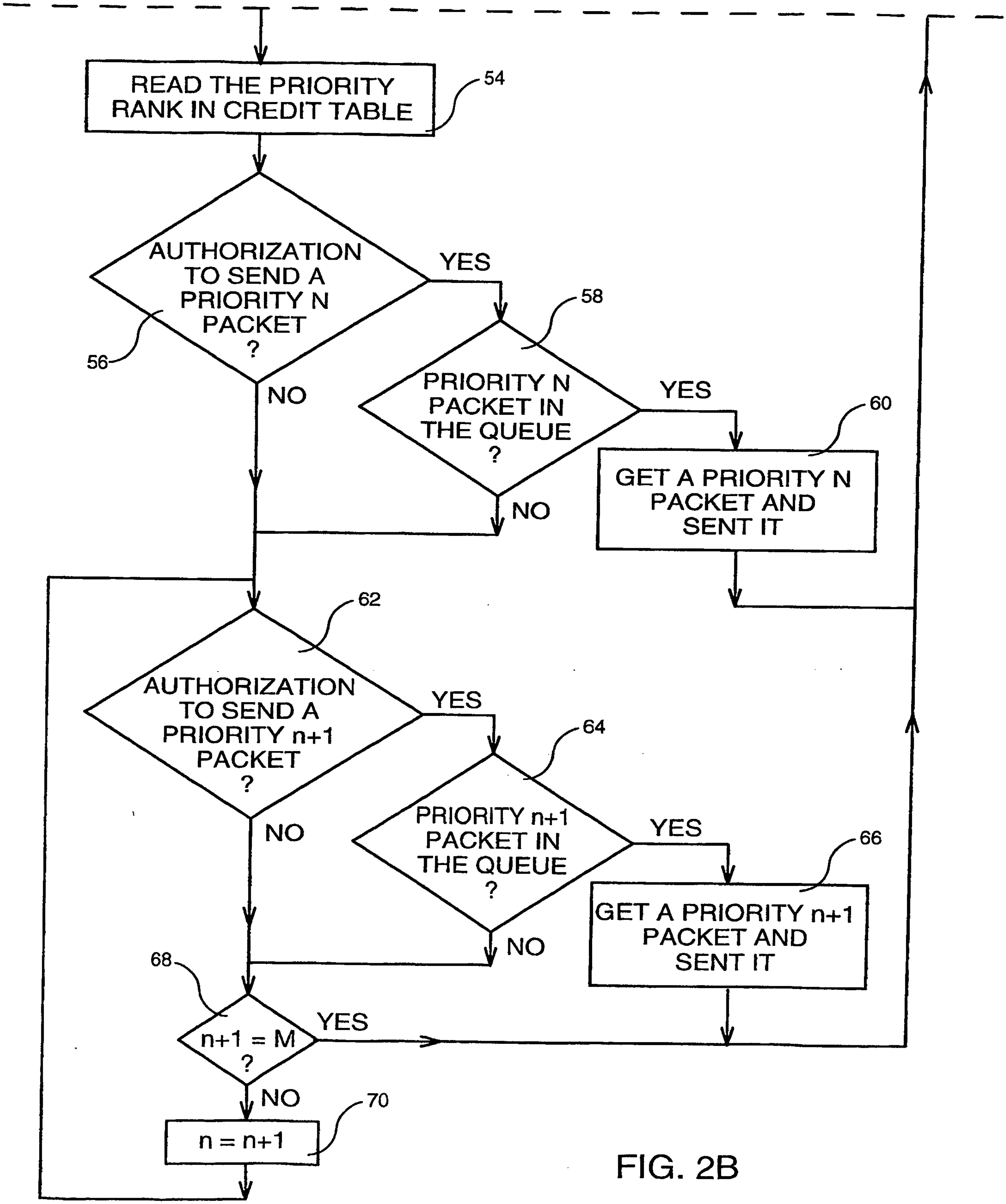
READ THE PRIORITY RANK IN CREDIT TABLE
54
AUTHORIZATION TO SEND A PRIORITY N PACKET ?
56
YES
NO
58
PRIORITY N PACKET IN THE QUEUE ?
YES
NO
60
GET A PRIORITY N PACKET AND SENT IT
62
AUTHORIZATION TO SEND A PRIORITY n+1 PACKET ?
YES
NO
64
PRIORITY n+1 PACKET IN THE QUEUE ?
YES
NO
66
GET A PRIORITY n+1 PACKET AND SENT IT
68
n+1 = M ?
YES
NO
70
n = n+1

FIG. 2B